



KEMENTERIAN RISET TEKNOLOGI DAN PENDIDIKAN TINGGI  
UNIVERSITAS BRAWIJAYA  
FAKULTAS TEKNIK  
JURUSAN TEKNIK ELEKTRO  
Jalan MT Haryono 167 Telp & Fax. 0341 554166 Malang 65145

**KODE  
PJ-01**

**PENGESAHAN  
PUBLIKASI HASIL PENELITIAN SKRIPSI  
JURUSAN TEKNIK ELEKTRO  
FAKULTAS TEKNIK UNIVERSITAS BRAWIJAYA**

**NAMA : HARI KURNIADI**  
**NIM : 125060309111006- 63**  
**PROGRAM STUDI : TEKNIK ELEKTRO**  
**JUDUL SKRIPSI : IMPLEMENTASI ALGORITMA KRIPTOGRAFI ELGAMAL  
UNTUK *FILE* CITRA 2 DIMENSI**

**TELAH DI-REVIEW DAN DISETUJUI ISINYA OLEH:**

Pembimbing 1

Pembimbing 2

**Waru Djuriatno, ST., MT.**  
**NIP. 19690725 199702 1 001**

**Adharul Muttaqin, ST., MT.**  
**NIP. 19760121 200501 1 001**

**IMPLEMENTASI ALGORITMA KRIPTOGRAFI ELGAMAL UNTUK  
FILE CITRA 2 DIMENSI**

**PUBLIKASI JURNAL SKRIPSI  
KONSENTRASI REKAYASA KOMPUTER**

Diajukan untuk memenuhi persyaratan  
memperoleh gelar Sarjana Teknik



Disusun Oleh :

**HARI KURNIADI**

**NIM : 125060309111006 - 63**

**KEMENTERIAN RISET DAN TEKNOLOGI PENDIDIKAN TINGGI  
UNIVERSITAS BRAWIJAYA  
FAKULTAS TEKNIK  
TEKNIK ELEKTRO  
MALANG  
2015**

# IMPLEMENTASI ALGORITMA KRIPTOGRAFI ELGAMAL UNTUK *FILE* CITRA 2 DIMENSI

HARI KURNIADI.<sup>1</sup>, Waru Djuriatno, ST., MT.<sup>2</sup>, Adharul Muttaqin, ST., MT.<sup>2</sup>  
<sup>1</sup>Mahasiswa Teknik Elektro Univ. Brawijaya, <sup>2</sup>Dosen Teknik Elektro Univ. Brawijaya

Jurusan Teknik Elektro Fakultas Teknik Universitas Brawijaya

Jalan MT. Haryono 167, Malang 65145, Indonesia

E-mail: hari.kurniadi.self@gmail.com

**Abstrak** – Secara umum ada dua tipe algoritma kriptografi berdasarkan kuncinya yaitu algoritma simetris dan algoritma asimetris. ElGamal adalah sebuah algoritma untuk kriptografi kunci publik atau asimetris. Algoritma asimetris terdiri atas dua buah kunci yaitu kunci publik untuk melakukan enkripsi sedangkan kunci pribadi untuk melakukan dekripsi. Algoritma asimetris terdiri atas dua buah kunci yaitu kunci publik untuk melakukan enkripsi sedangkan kunci pribadi untuk melakukan dekripsi.

Algoritma ElGamal biasanya digunakan untuk enkripsi data berupa teks saja. Skripsi ini menguraikan bagaimana menerapkan algoritma ElGamal pada file citra 2 dimensi. Algoritma ini menggunakan beberapa persamaan untuk melakukan proses generate key, proses enkripsi dan proses dekripsi. Citra yang akan di enkripsi ke algoritma ini adalah plain citra grayscale. Modifikasi yang dilakukan adalah dengan mengubah kunci public  $g$  dan  $z$  menjadi representasi data 2 dimensi, dengan  $g$  merupakan citra grayscale dan  $z$  adalah citra RGB. Hasil akhir menunjukkan bahwa enkripsi berhasil dilakukan dengan nilai koefisien korelasi antara citra plain dan chipper berada pada nilai kurang dari satu.

**Kata Kunci** — Kriptografi, Enkripsi Citra, ElGamal, Digital Image Processing, Grayscale.

## I. PENDAHULUAN

Masalah keamanan dan kerahasiaan data merupakan suatu hal yang sangat penting terutama di dalam menghadapi persaingan bisnis. Data yang bersifat rahasia perlu dibuatkan sistem penyimpanan dan pengirimannya agar tidak terbaca atau diubah oleh pihak yang tidak berwenang, baik saat data tersebut tersimpan sebagai *file* di dalam komputer maupun saat data tersebut dikirim melalui media Internet seperti *email*, *ftp*, dan media penyimpanan online lainnya. *File* citra digital atau gambar terkadang merupakan suatu aset berharga. Misalkan suatu hasil pesanan desain yang masih tahap pengembangan yang perlu ditunjukkan kepada calon pembeli ataupun gambar atau foto yang bersifat pribadi dan rahasia.

Ada berbagai macam cara atau metode untuk mengamankan suatu *file* atau data. Salah satunya adalah kriptografi. Kriptografi merupakan metode untuk mengamankan data, baik itu data teks maupun data gambar. Metode ini dilakukan dengan penyandian atau pengacakan data asli, sehingga pihak lain yang tidak

mempunyai hak akses atas data tersebut tidak dapat memperoleh informasi yang ada di dalamnya.[1]

Enkripsi citra merupakan teknik untuk melindungi kerahasiaan citra dari pengaksesan ilegal. Enkripsi diperlukan karena dalam era digital sekarang ini citra digital mudah disimpan atau ditransmisikan melalui saluran publik seperti internet. Pengiriman citra melalui saluran publik rawan terhadap penyadapan, dan penyimpanan citra di dalam media storage rawan terhadap pengaksesan oleh pihak-pihak yang tidak memiliki otoritas. Enkripsi menyandikan citra (*plain-image*) ke bentuk visual lain yang tidak bermakna (*cipher-image*).

Secara umum ada dua tipe algoritma kriptografi berdasarkan kuncinya yaitu algoritma simetris dan algoritma asimetris. Algoritma simetris adalah algoritma yang memiliki kunci enkripsi dan dekripsi yang sama, sedangkan untuk algoritma asimetris terdiri atas dua buah kunci yaitu kunci publik untuk melakukan enkripsi sedangkan kunci pribadi untuk melakukan dekripsi. Dalam algoritma kunci asimetris, kunci yang didistribusikan adalah kunci publik yang tidak diperlukan kerahasiaannya sedangkan kunci pribadi tetap disimpan atau tidak didistribusikan. Setiap orang yang memiliki kunci publik dapat melakukan proses enkripsi tetapi hasil dari enkripsi tersebut hanya bisa dibaca oleh orang yang memiliki kunci pribadi. [6]

Algoritma ElGamal adalah sebuah algoritma untuk kriptografi kunci publik. Algoritma ini dibuat oleh Taher ElGamal pada tahun 1985. Algoritma ini memiliki keamanan yang terletak pada kesulitan dalam menghitung logaritma diskrit. [2] Pada umumnya algoritma ElGamal hanya untuk data berupa teks saja. Namun penulis akan mencoba menerapkan algoritma tersebut untuk data 2 dimensi atau citra.

Algoritma kriptografi ElGamal menggunakan beberapa persamaan untuk melakukan proses generate key, proses enkripsi dan proses dekripsi. Citra yang akan di implementasikan ke algoritma ini adalah citra grayscale. Mengubah *file* citra menjadi blok angka berdasarkan nilai intensitas.

## II. TINJAUAN PUSTAKA

### A. Kriptografi

kriptografi adalah ilmu yang mempelajari teknik-teknik matematika yang berhubungan dengan aspek keamanan informasi seperti kerahasiaan data, keabsahan data, integritas data, serta autentikasi data. Namun, tidak semua aspek keamanan informasi

ditangani oleh kriptografi. Enkripsi erat kaitannya dengan dekripsi, untuk itulah muncul istilah kriptanalisis. Kriptanalisis adalah ilmu dan seni untuk memecahkan informasi yang telah dienkripsi tanpa mengetahui kunci yang digunakan. Pelaku kriptanalisis disebut dengan kriptanalisis.

### Prinsip Kriptografi

Pada prinsipnya, Kriptografi memiliki 4 komponen utama yaitu:

1. *Plain* citra, yaitu pesan yang dapat dibaca
2. *Cipher* citra, yaitu pesan acak yang tidak dapat dibaca
3. Key, yaitu kunci untuk melakukan teknik kriptografi
4. Algorithm, yaitu metode untuk melakukan enkripsi dan dekripsi

Adapun proses dasar pada Kriptografi yaitu:

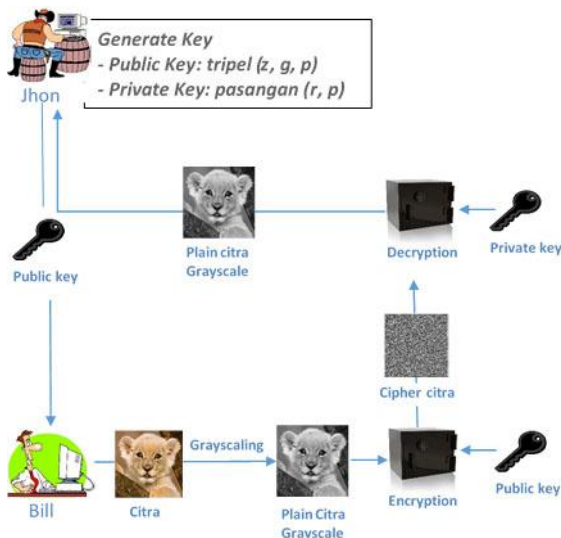
1. Enkripsi (Encryption)
2. Dekripsi (Decryption)

### B. Algoritma ElGamal

Algoritma ElGamal adalah sebuah algoritma untuk kriptografi kunci public. Algoritma ini dibuat oleh Taher ElGamal pada tahun 1985. Algoritma ini memiliki keamanan yang terletak pada kesulitan dalam menghitung logaritma diskrit.

Algoritma kriptografi ElGamal menggunakan beberapa persamaan untuk melakukan proses generate key, proses enkripsi dan proses dekripsi.

#### Blok Diagram Sistem



Gambar 1 Blok Diagram Sistem

Keterangan gambar 1 :

Bill akan mengirim citra kepada Jhon. Jhon membuat 2 buah kunci yaitu kunci publik dan kunci privat dan akan memberikan kunci publiknya kepada Bill. Kemudian Bill mengubah citra berwarna menjadi citra abu-abu (grayscale). Setelah itu Bill mengenkripsi

menggunakan kunci publik algoritma ElGamal menjadi *ciphercitra* dan mengirim kepada Jhon.

Jhon menerima *ciphercitra* dan akan mendekripsi menggunakan kunci privat algoritma ElGamal sehingga Jhon mendapatkan *plain* citra grayscale dari Bill.

#### Proses Generate Key

Algoritma ElGamal memerlukan sepasang kunci yang dibangkitkan dengan memilih bilangan prima  $p$  dan dua buah bilangan acak (random)  $g$  dan  $r$  dengan syarat  $g < p$  dan  $1 \leq r \leq p - 2$  yang memenuhi persamaan.

$$z = g^r \text{ mod } p$$

Dari persamaan tersebut nilai  $z$ ,  $g$  dan  $p$  merupakan pasangan kunci publik sedangkan  $r$ ,  $p$  merupakan pasangan kunci pribadi.

Properti algoritma ElGamal sebagai berikut:

1. Bilangan prima,  $p$  (tidak rahasia)
2. Bilangan acak,  $g$  ( $g < p$ , grayscale) (tidak rahasia)
3. Bilangan acak,  $r$  ( $r < p$ , kunci privat) (rahasia)
4.  $z = g^r \text{ mod } p$  (kunci publik,RRGGBB) (tidak rahasia)
5.  $m$  atau  $f(x,y)$  (*plain* citra, grayscale) (rahasia)
6.  $a$  dan  $b$  ( $a$  dan  $b$  adalah *ciphercitra*,RRGGBB) (tidak rahasia)

#### Proses Enkripsi

Pada proses enkripsi dilakukan dengan menyusun nilai-nilai intensitas sesuai blok-blok pada *pixel* citra. Nilai-nilai ini yang disebut nilai  $m$  (*plain* citra). nilai  $m$  harus masih berada didalam range 0 sampai  $p - 1$ . Kemudian memilih bilangan acak  $k$ , yang dalam hal ini  $1 \leq k \leq p - 2$ . Setiap blok dienkripsi dengan rumus.

$$a = g^k \text{ mod } p$$

$$b = z^k m \text{ mod } p$$

#### Proses Dekripsi

Pada proses dekripsi digunakan kunci pribadi  $r$  dan  $p$  untuk mendekripsi  $a$  dan  $b$  menjadi *plain* citra  $m$  dengan persamaan.

$$(a^r)^{-1} = a^{p-1-r} \text{ mod } p$$

$$m = b/a^r \text{ mod } p = b(a^r)^{-1} \text{ mod } p$$

### C. Pengubahan Citra Berwarna menjadi Citra Grayscale

Citra berwarna diubah menjadi citra grayscale dengan mengubah format citra yang awalnya adalah RGB menjadi YUV lalu diambil Y-nya saja. Secara matematis, perhitungan citra grayscale akan menjadi :  
Keterangan : R = Merah, G = Hijau, B = Biru,

$$\text{Grayscale} = 0.299R + 0.587G + 0.114B$$

Dengan cara ini, maka citra berwarna akan berubah menjadi grayscale tanpa mengubah keaslian dari RGB.  
[4]

## III. PERANCANGAN DAN IMPLEMENTASI PERANGKAT LUNAK

Perancangan dan implementasi dikerjakan dengan beberapa tahap. Meliputi pembangkitan kunci privat dan kunci publik, proses enkripsi, dan proses dekripsi.

### Representasi Pixel

Untuk mengimplementasikan algoritma ElGamal, perlu diperhatikan representasi *pixel* yang dienkripsi maupun didekripsi. Untuk algoritma tersebut menggunakan representasi *pixel* yang sama. Sebuah citra berwarna merupakan objek 2D yang tersusun dari *pixel-pixel*. Proses enkripsi dan dekripsi dengan cara menelusuri setiap *pixel* secara iteratif sehingga dapat mengakses seluruh *pixel*. Kemudian untuk setiap *pixel* di dapat warna yang di representasikan sebagai RGB masing masing 1 byte. Sebagai contoh, warna putih total memiliki nilai red=255, green=255, dan blue=255.

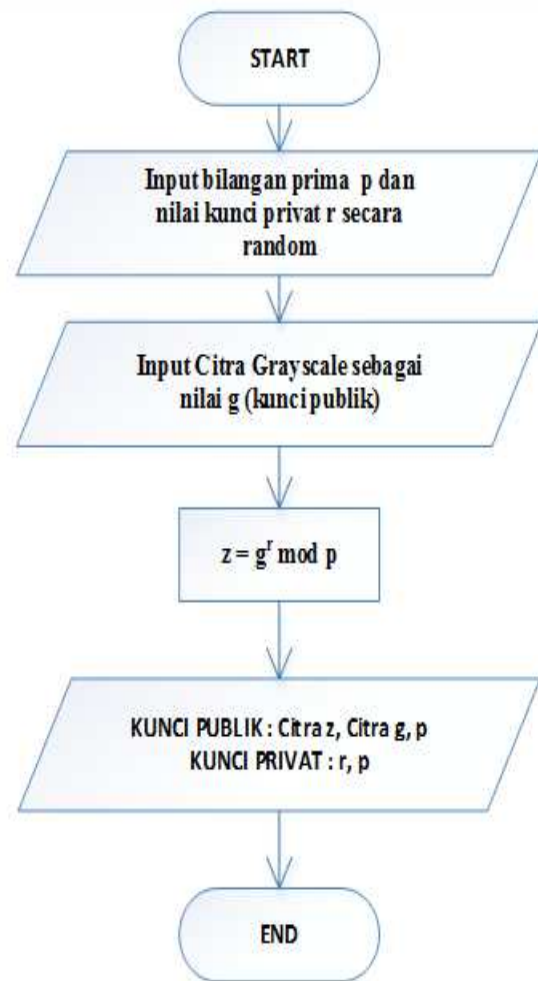
Agar *pixel* yang diakses dapat di terapkan sebagai message maka dari *pixel* yang didapat dibentuk menjadi sebuah bilangan integer. Bilangan yang dimaksud didapat dari mengubah nilai red, green, dan blue pada sebuah *pixel* menjadi hexadesimal 8 bit untuk setiap nilainya. Masing masing nilai kemudian di append 1 sama lain dimulai dari red, green, kemudian blue. Setelah mendapatkan hexadesimal 24 bit maka segera dikonversikan menjadi bilangan integer, bilangan inilah yang nantinya menjadi message untuk diterapkan dalam sebuah algoritma ElGamal.

Sebagai contoh diberikan sebuah citra berwarna dengan ukuran 512 x 512 *pixel*. Kemudian diambil *pixel* ke (0,0) dan didapat *pixel* dengan warna yang putih (red=255,green=255,blue=255). Masing – masing di konversi menjadi hexadesimal 8 bit dan nilainya menjadi red=FF,green=FF, dan blue=FF. Ketiga hexadesimal yang didapat kemudian digabung menjadi satu dimulai dari red, green, blue dan menjadi *pixel* =FFFFFF. Hexadesimal *pixel* lalu dikonversi menjadi bilangan integer. *Pixel* putih mempunyai integer = 16777215. Bilangan tersebut yang nantinya sebagai message yang akan di proses. Cara sebaliknya dapat digunakan untuk membentuk sebuah *pixel* berwarna dari hasil enkripsi.

Pada prinsipnya proses dekripsi melakukan hal yang sama dengan ketika mengenkripsi sebuah *pixel*, yang membedakan adalah algoritma dekripsinya saja. *Pixel* yang ingin di dekripsi harus diubah dalam bentuk yang dapat di masukkan ke dalam algoritma kemudian hasilnya dibentuk ulang menjadi sebuah *pixel* berwarna atau grayscale. [8]

### Proses Generate Key

Rancangan proses generate key dapat dilihat pada Gambar 2.



Gambar 2 Flowchart Proses Generate Key

Pada proses ini dilakukan untuk menentukan nilai kunci yang meliputi variabel p, g, r, dan z. Dimana nilai variabel p adalah nilai bilangan prima random. Variabel r adalah nilai random integer dengan ketentuan  $1 < r < p-2$ . g dan z adalah nilai *pixel* suatu citra seperti pada persamaan  $z = g^r \text{ mod } p$ .

### Proses Enkripsi




Proses enkripsi merupakan proses untuk mengubah data sumber menjadi *file cipher* citra dengan menggunakan nilai-nilai kunci publik yang dihasilkan dari proses *generate key*. Rancangan proses enkripsi dapat dilihat pada gambar 3.

#### IV. PENGUJIAN DAN PEMBAHASAN

##### A. Berkas Citra Uji





Berikut adalah berkas citra yang akan di uji pada tabel 1.

**Tabel 1** Berkas Citra Uji

No.	Plain Citra	Ukuran	Kunci Publik Citra g	Ukuran	Dimensi	Format
1	 baboon	192 KB	 goldhill	65 KB	200 x 200	bmp
2	 peppers	768 KB	 boat	257 KB	500 x 500	bmp

##### B. Pengujian Proses *Generate Key*







**Tabel 2** Hasil Uji Proses *Generate Key*

No.	Citra g	Ukuran	Citra z	Ukuran	Dimensi	Format
1	 goldhill	65.0 KB		256 KB	200 x 200	bmp
2	 boat	257 KB		1.00 MB	500 x 500	bmp

Dari hasil uji tabel 2, hasil citra z terlihat sedikit sama dengan citra g dan mempunyai ukuran yang lebih besar dari citra g.

##### C. Pengujian Proses Enkripsi









**Tabel 3** Hasil Uji Proses Enkripsi

No.	Plain Citra	Cipher A	Ukuran	Cipher B	Ukuran	Dimensi
1.	 baboon		256 KB		256 KB	200 x 200
2.	 peppers		1.00 MB		1.00 MB	500 x 500

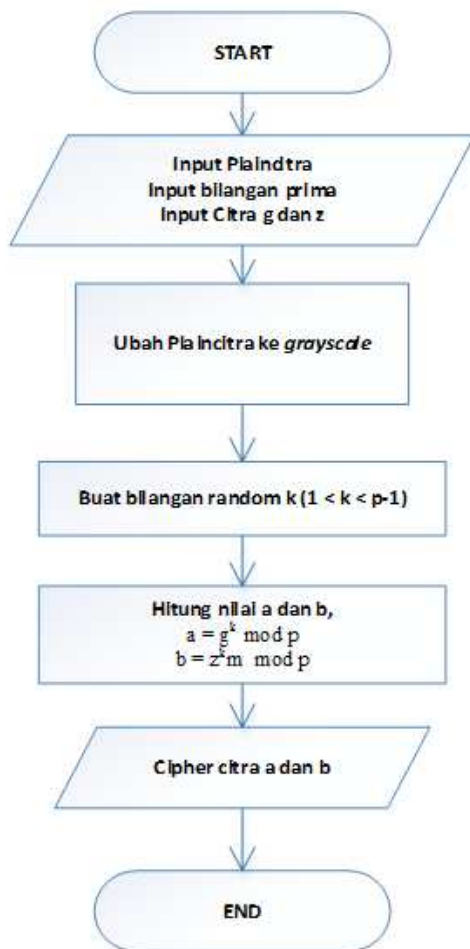
Dari hasil uji tabel 3, hasil *cipher* citra mempunyai ukuran 2 kali dari *plain* citra. Dan untuk *cipher* A masih terlihat sedikit sama dengan citra z. sedangkan *cipher* B terlihat lebih acak.

##### D. Pengujian Proses Dekripsi

**Tabel 4** Hasil Uji Proses Dekripsi

No.	Plain Citra	Cipher A	Cipher B	Hasil Plain Citra	Ukuran Hasil Plain Citra	Dimensi
1.	 baboon				256 KB	200 x 200
2.	 peppers				1.00 MB	500 x 500

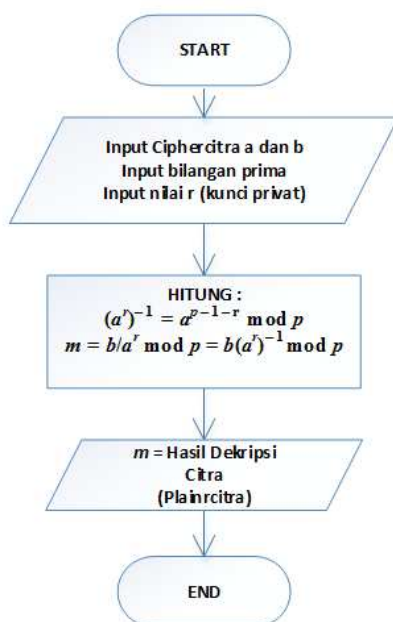
Dari hasil uji tabel 4, terlihat hasil *plain* citra dapat ditemukan kembali. Dan ukuran hasil *plain* citra lebih besar dari ukuran *plain* citra awal.



Gambar 3 Flowchart Proses Enkripsi

##### Proses Dekripsi

Proses dekripsi adalah proses untuk mengembalikan *cipher* citra kedalam bentuk *plain* citra, dengan menggunakan kunci privat (r,p). Rancangan proses dekripsi dapat dilihat pada gambar 4.



Gambar 4 Flowchart Proses Dekripsi









## E. Analisis Hasil Plain Citra dan Cipher Citra

Pada bagian ini dilakukan 2 metode analisis, yaitu analisis korelasi dan analisis histogram.

### Analisis Korelasi

Di dalam *natural-image*, *pixel-pixel* yang bertetangga memiliki hubungan linier yang kuat. Ini ditandai oleh koefisien korelasinya yang tinggi (mendekati +1 atau -1). Di dalam citra acak, korelasi antar *pixel* bertetangga tidak ada atau koefisien korelasinya nol. Enkripsi citra bertujuan membuat korelasi *pixel-pixel* yang bertetangga di dalam *cipher-image* menjadi lemah atau dengan kata lain membuat koefisien korelasinya mendekati nol. Untuk mengetahui korelasi *pixel-pixel* di dalam *plain-image* maupun *cipher-image*, maka dihitung koefisien korelasi antara dua *pixel* bertetangga secara horizontal [ $f(i,j)$  dan  $f(i, j+1)$ ], dua *pixel* bertetangga secara vertikal [ $f(i,j)$  dan  $f(i+1, j)$ ], dan dua *pixel* bertetangga secara diagonal [ $f(i,j)$  dan  $f(i+1, j+1)$ ]. Secara acak dipilih 1000 pasang *pixel* bertetangga pada setiap arah (vertikal, horizontal, dan diagonal), masing-masing pada citra *plainimage* dan *cipher-image*. Tanpa kehilangan generalisasi, analisis korelasi dilakukan pada citra grayscale saja. Dalam hal ini x dan y adalah nilai keabuan dari dua *pixel* bertetangga. [7] Berikut hasil perhitungan korelasi pada tabel 5.

**Tabel 5** Perbandingan Koefisien Korelasi antara dua *pixel* bertetangga

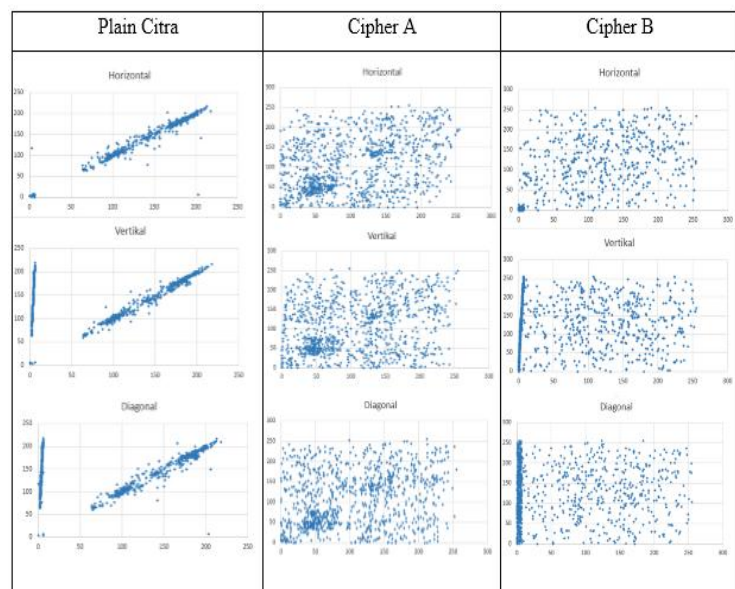
Citra	Korelasi	Citra	Korelasi
	Horizontal 0.548474 Vertikal 0.533306 Diagonal 0.394014		Horizontal 0.99338 Vertikal 0.262402 Diagonal 0.251838
	Horizontal 0.405652 Vertikal 0.311766 Diagonal 0.206862		Horizontal 0.357317 Vertikal 0.248607 Diagonal 0.239924
	Horizontal 0.131774 Vertikal 0.191518 Diagonal 0.027682		Horizontal 0.704049 Vertikal 0.051994 Diagonal 0.029243

Dari tabel 5 dapat dilihat bahwa koefisien *plain* citra lebih mendekati 1 daripada koefisien *cipher* citra, yang mengindikasikan *plain* citra

mempunyai korelasi yang lebih kuat dari pada *cipher* citra. Pada koefisien *cipher* A masih terlihat adanya sedikit korelasi. Sedangkan *cipher* B koefisien korelasi mendekati 0 yang berarti *pixel-pixel* yang bertetangga tidak lagi berkorelasi.

Untuk melihat lebih jelas korelasi antara *pixel-pixel* bertetangga, maka gambar 5 memperlihatkan distribusi korelasi *pixel-pixel* yang bertetangga. Pada *plain* citra dapat dilihat bahwa *pixel-pixel* yang bertetangga nilai-nilainya saling berkorelasi. Sedangkan pada *cipher* citra A dan B nilainya tersebar dan sedikit nilai yang saling berkorelasi.

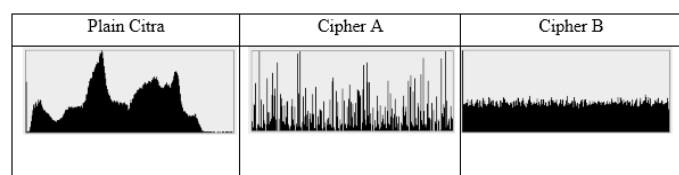
Gambar 5 Distribusi Korelasi *Pixel-pixel* Bertetangga Pada Plain Citra dan Cipher Citra dari Citra 'peppers'



### Analisis Histogram

Di dalam bidang pengolahan citra histogram memperlihatkan distribusi nilai *pixel* di dalam sebuah citra. Histogram digunakan penyerang (*attacker*) untuk melakukan kriptanalisis dengan memanfaatkan frekuensi kemunculan *pixel* di dalam histogram. Penyerang berharap nilai *pixel* yang sering muncul di dalam *plain-image* berkorelasi dengan nilai *pixel* yang sering muncul di dalam *cipher-image*. Dengan menganalisis frekuensi kemunculan nilai *pixel*, penyerang mendeduksi kunci atau *pixel-pixel* di dalam *plain-image*.

Agar penyerang tidak dapat menggunakan histogram untuk melakukan analisis frekuensi, maka histogram *plain-image* dan histogram *cipherimage* seharusnya berbeda secara signifikan atau secara statistik tidak memiliki kemiripan. Oleh karena itu, histogram *cipher-image* seharusnya datar (*flat*) atau secara statistik memiliki distribusi (relatif) *uniform*. Distribusi yang (relatif) *uniform* pada *cipher-image* adalah sebuah indikasi bahwa algoritma enkripsi citra memiliki tingkat keamanan yang bagus. [7]



Gambar 6 Histogram Citra 'peppers'

Gambar 6 memperlihatkan histogram pada *cipher* B terlihat datar (*flat*) atau terdistribusi *uniform* dan berbeda signifikan dengan histogram *plain* citra dan *cipher* A.

## V. PENUTUP

### A. Kesimpulan

Dari proses perancangan, implementasi dan pengujian program *generate key*, enkripsi dan dekripsi didapat kesimpulan antara lain.

1. Algoritma ElGamal dapat diterapkan untuk citra 2 dimensi.

2. Perangkat lunak yang mengimplementasikan algoritma ElGamal untuk citra 2 dimensi berhasil dibangun dengan menerapkan kunci publik  $g$  dan  $z$  sebagai citra 2 dimensi, dimana  $g$  adalah citra *grayscale* dan  $z$  adalah citra RGB.
3. Analisis korelasi memperlihatkan *pixel-pixel* di dalam *cipher-image* tidak berkorelasi satu sama lain (memiliki koefisien korelasi yang mendekati nol), sehingga algoritma aman dari serangan analisis statistik untuk menemukan kunci atau *plain-image*.
4. Analisis histogram memperlihatkan bahwa histogram *cipher-image* berbentuk datar atau terdistribusi *uniform*, sehingga algoritma aman dari serangan analisis frekuensi.

## B. Saran

Saran yang dapat diberikan adalah :

1. Mengganti kunci citra  $g$  dengan citra yang lebih acak atau citra tidak bermakna.
2. Jika ukuran dimensi citra  $g$  dan  $z$  lebih kecil dari ukuran *plain image*, maka pada citra  $g$  dan  $z$  akan melakukan perulangan citra sampai sama dengan ukuran *plain image*.
3. Perlu adanya analisis lebih lanjut untuk penerapan algoritma ini seperti analisis entropi, analisis sensitivitas, dan lain-lain.

## DAFTAR REFERENSI

- [1] Slide kuliah Rinaldi Munir IF3058 Kriptografi
- [2] Munir, Rinaldi, 2005. Algoritma ElGamal, STEI – ITB
- [3] Munir, Rinaldi, 2006. Algoritma RSA dan ElGamal. IF-ITB .
- [4] Munir, Rinaldi. Pengolahan Citra Digital Dengan Pendekatan Algoritmik. Informatika, Bandung. 2008.
- [5] Sutoyo, T., Mulyanto, Edy., Suhartono, Vincent., Nurhayati, Oky Dwi., Wijanarto. 2009. Teori Pengolahan Citra Digital. Yogyakarta: Andi.
- [6] Rashmi Singh, Shiv Kumar, “ElGamal’s Algorithm in Cryptography”, International Journal of Scientific & Engineering Research, 2012, Volume 3.
- [7] Munir, Rinaldi, “Analisis Keamanan Algoritma Enkripsi Citra Digital Menggunakan Kombinasi Dua Chaos Map Dan Penerapan Teknik Selektif”, Juti, 2012, Volume 10.
- [8] Chaq, Whilda, “Perbandingan Algoritma Kunci Nirsimetris ElGamal dan RSA pada Citra Berwarna”, IF3058 Kriptografi Sem. II, 2012/2013.